Closed-Form Expression For The Resources Dimensioning of Softwarized Network Services

Jonathan Prados-Garzon[*], Tarik Taleb[*§], Oussama El Marai[*], and Miloud Bagaa[*]
jonathan.prados-garzon@aalto.fi, tarik.taleb@aalto.fi, oussama.elmarai@aalto.fi, miloud.bagaa@aalto.fi
[*]Aalto University, Espoo, Finland.
[§]University of Oulu, 90570 Oulu, Finland.

***Abstract*—Network Function Virtualization (NFV) ecosystem enables the automation of deployment and scaling of softwarized network services (SNSs), thus reducing their operational expenditures. This enables operators to handle workload fluctuations, to keep the desired performance, with great agility and reduced costs. However, to realize the automation of such management practices, it is needed to determine the amount of required resources to allocate the SNS so that its performance requirements are met. This problem is commonly referred to as resources dimensioning problem. In this paper, we address the derivation of a closed-form expression for the optimal resources dimensioning of an SNS in terms of cost or energy efficiency. The performance requirement considered for the SNS is a limit on its mean response time. The performance model considered for the SNS is practical and accurate. The usefulness of the derived closed-form expression is successfully validated by means of simulation. The scenario considered for the validation is a video optimization chain located at the SGi-LAN of a mobile network.**

I. Introduction

Network Functions Virtualisation (NFV) paradigm is envisaged as a cornerstone to build future networks. NFV decouples network functions (e.g., firewalling, load balancing, mobility management, deep packet inspection, etc.) from proprietary hardware enabling them to run as software components, which are called Virtual Network Functions (VNFs), on virtualization containers (e.g., Virtual Machines (VMs) and OS-level containers) [1]. Among its benefits, NFV promises to enable network operators the automation of the management operations and orchestration of the future networks, thus reducing the Operating Expenditures (OPEXs) and accelerating time-to-market of new services [2]–[5].

Particularly notable among the envisioned management practices facilitated by NFV are the automation of deployment and scaling of network services [5], [6]. This is thanks to VNFs can be instantiated on-demand and at different network locations without requiring on-site personnel to deploy new hardware as needed traditionally. In this way, the resources allocated to the different network services can be automatically increased or decreased, thus enabling network operators to handle workload fluctuations to keep the desired performance with great agility and efficiency while reducing the total cost. However, to realize such a scenario, it is required to define solutions to determine when and how much resources have to be provisioned to a given network service so that the target performance metrics be always met. This problem is typically known in the literature as Dynamic Resource Provisioning (DRP).

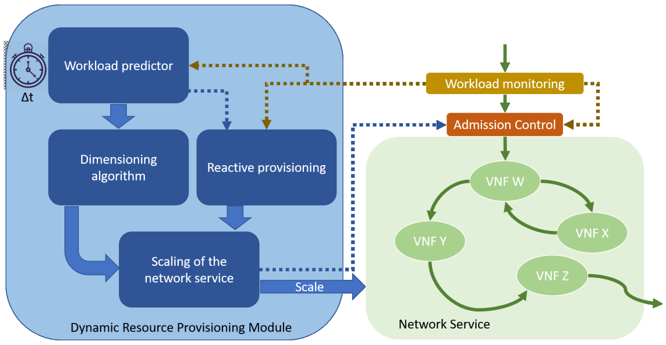

Fig. 1. DRP solution for a network service.

Figure 1 depicts a possible DRP solution for network services. A similar architecture was considered for the DRP of Internet applications with successful results in [7]. It combines proactive and reactive provisioning mechanisms. The proactive mechanism is executed synchronously every Δt units of time. Conversely, the reactive provisioning might be run asynchronously when it detects the workload predictor made a significant prediction error. In either case, the dimensioning module will be invoked to estimate the amount of required resources for a given workload so that the performance requirements are ensured.

In this context, the present work derives a closed-form expression for the optimal resources dimensioning of a Softwarized Network Service (SNS) in terms of cost or energy efficiency. The performance requirement considered for the SNS is a limit on its mean response time. To that end, we consider a practical and accurate performance model for the SNS. Under this consideration, we show that the formulated problem of a network service is convex and find its explicit solution by using the method of Lagrange multipliers. The usefulness of the derived closed-form expression is successfully validated through simulation. The scenario considered for the validation is a video optimization chain located at the SGi-LAN of a mobile network.

The remainder of the paper is organized as follows. Section II reviews the related literature. Section III includes the system model and formulation of the resources dimensioning problem of an SNS. Section IV describes a simple but practical and accurate performance model for SNSs. Section V contains the closed-form expression to perform the resources dimensioning of an SNS. Section VI describes experimentation carried out and the achieved results which verify the usefulness of the derived expression. Finally, Section VII concludes the paper.

II. Related Works

The DRP solutions can be broadly categorized into rule-based and model-based approaches [8]. The rule-based approaches, such as those proposed in [9] and [10], are based on reinforcement learning, statistical machine learning, and fuzzy control. On the other hand, the model-based approaches are based on control theory and Queueing Theory (QT). Compared to rule-based approaches, model based approaches require more domain knowledge, but can provide Quality of Service (QoS) guarantees, while ensuring the system stability [8]. Here we will focus on the resources dimensioning, which is a paramount component of DRP solutions, of the softwarized network services following a model-based approach [11]–[20].

In [16], [17], the authors employ a Jackson's network to model a three-tiered virtualized Mobility Management Entity (vMME). They use an exhaustive search methodology to perform the dimensioning of the number of vMME worker instances. However, the jointly dimensioning of many resources following a brute force approach is expensive in terms of computational effort. In [21], the authors provide a useful simple model based on time series to predict the computational resources demand in the Evolved Packet Core (EPC).

There are several heuristics proposed in the literature to tackle the Resources Dimensioning (RD) problem of softwarized network services [11], [13], [18], [20]. In [11], the authors formulate and propose a heuristic to solve the joint optimization problem for the Service Function Chain (SFC) routing and VNF instance dimensioning. The objective of the problem is to maximize the number of accepted SFC requests. In [13], the authors formulate the RD problem to minimize the expected waiting time of service chains. The authors employ a mixed multi-class Baskett, Chandy, Muntz, and Palacios (BCMP) network to model a service chain and solve it by using the Mean Value Analysis (MVA) algorithm. Although the authors prove the convexity of the problem, its solution cannot be found because of the required closed queuing network calculation. Then, the authors propose a heuristic method to address the issue. In [18], [20], the authors propose a heuristic to carry out the joint RD of the Control Plane (CP) entities of a virtualized EPC. The heuristic requires an auxiliary methodology to predict the vEPC response time for a given setup. To that end, the authors propose a holistic QT-based model for the Long Term Evolution (LTE) CP. They evaluate their heuristic proposal in the context of planning [20] and DRP [18].

There are also examples of the use of metaheuristics to find a solution to the RD problem in the context of NFV. In [19], the authors propose a genetic algorithm to solve the RD problem. The algorithm tries to minimize the service blocking rate and CPU usage in Cloud/Mobile Edge Computing (MEC) Radio Access Network (RAN)-based 5G architectures.

Finally, some works find the optimal solution for the RD of the softwarized network services problem [12], [14], [22], though they are tailored for specific use cases. In [12], the authors formulate the RD problem of a Content Delivery Network (CDN). Notably, they aim at minimizing the amount of resources (virtual CPUs) under capacity constraints so that a given QoE for the end user is met. To solve this problem, they propose a novel algorithm. In [22], the authors develop a bi-class (e.g., machine-to-machine -M2M- and mobile broadband -MBB- communications) queuing model for the vEPC. The CP and Data Plane (DP) of the vEPC are respectively modeled as M/M/m/m and M/D/1 nodes. The authors assume that the Mobility Management Entity (MME) and Serving Gateway (SGW)/Packet Data Network Gateway (PGW) nodes run on the same Physical Machine (PM). They formulate and solve the problem of distributing the PM resources among the MME and PGW nodes in order to minimize the blocking rate of M2M sessions. The authors in [23] investigate the fairness-aware flow scheduling problem for network utility maximization. As part of this work, the authors model the computational resources demanded by the flows in a chain of VNFs. In [14], the authors propose a model for sizing a Cloud-RAN infrastructure. More precisely, they suggest and validate by means of simulation the bulk arrival model $M^{[X]}/M/C$ to predict the processing time of a subframe in a Cloud-RAN architecture based on multi-core platform. Based on this model, the authors compute the required number of cores C to be allocated to the Cloud-RAN as the minimum value of C so that $P[T > \delta] < \varepsilon$, i.e., the probability that the subframe processing time exceeds a given value is acceptable. In the same context, the authors in [24] propose a base station agnostic framework for creating wireless slices in a cellular RAN. As part of this work, the authors provide a model to estimate the processing load of the Baseband Unit pool.

III. System model and problem formulation

Let us consider a Softwarized Network Service (SNS) as an arbitrary composition of VNFs. Each VNF, in turn, might consist of one or several Virtual Network Function Components (VNFCs), each of which provides part of the VNF functionality, working together. Each VNFC might have several instances, each of which runs as a software component in an isolated Virtualization Container (VC) (e.g., VM or OS container). The packets enter and leave the SNS through its external interfaces. The different packet flows served by the SNS may follow any arbitrary path across the VNFCs. Here, we assume that the load is distributed among the instances of a given VNFC in accordance with their computational capacities.

The PMs that host the VCs are interconnected through a set of network devices. The packets consume resources (e.g., CPU, RAM, Disk I/O, network I/O, etc.) of both PMs and network devices during its lifetime in the SNS. We will consider that VNFCs execute CPU-intensive tasks for processing the packets. Under this assumption, the CPU is the resource acting as the main bottleneck at the VNFCs instances. In this context, the resource dimensioning problem of the different SNS VNFCs is to determine how many CPU cores have to be allocated to the distinct VNFCs so that a set of performance requirements are met given an SNS workload. Next, we will formulate formally this problem considering that the mean response time of the SNS $\overline{T}$ has to be bounded (i.e., $\overline{T} \leq \overline{T}_{max}$) as the only performance requirement.

Let us assume there are J different VNFCs that make up the SNS and let m_j denote the number of CPU cores to be allocated to the VNFCs $j \in [1,J] \cap \mathbb{N}$ so that $\overline{T} \leq T_{max}$. The VNFC resource dimensioning problem is formally formulated as follows:

$$minimize \left(\sum_{j=1}^{J} \alpha_j \cdot m_j \right) \tag{1}$$

where α_j is a cost associated with the processing instances allocated to the VNFC j (to make the problem more generic).

Subject to :

$$C1: \quad \overline{T} \leq \overline{T}_{max} \tag{2}$$

The decision variables of the optimization problem are $m_j \forall j \in [1,J]$. Objective (1) intends to minimize the economic cost or the energy consumption depending on the meaning of α_j. Constraint (2) ensures that the SNS mean delay $\overline{T}$ is below a given threshold $\overline{T}_{max}$.

IV. Performance Model for SNSs

This section describes the performance model considered for the SNSs. The model is based on QT and allows us to analytically estimate the mean response time of an SNS $\overline{T}$. $\overline{T}$ is formally defined as the expected delay experienced by an arbitrary job during its stay in the SNS. A job might be a single packet or a set of packets depending on the specific scenario.

The pool of CPU cores allocated to the VNFC j is modeled as a set of m_j parallel $G/G/1$ queues as in [7]. The service process of each CPU core (or $G/G/1$ node) is described by its mean service rate μ_j and Squared Coefficient of Variation (SCV) of the service time c_{sj}^2. The aggregated arrival process to the pool of CPU cores of the VNFC j is described by the mean arrival rate λ_j and SCV of the inter-arrival times c_{aj}^2. These parameters (μ_j, c_{sj}^2, λ_j, and c_{aj}^2) are assumed to be known. In practice, this can be realized by using monitoring and predictive techniques. The workload of a given VNFC is distributed equally among its CPU cores. The mean response time of the pool of CPU cores of the VNFC j, $\overline{T}_j$, is approximated as:

$$\overline{T}_j = \frac{c_{sj}^2 + c_{aj}^2}{2} \cdot \frac{\rho_j}{\mu_j \cdot (m_j - \rho_j)} + \frac{1}{\mu_j} \tag{3}$$

where $\rho = \lambda_j / \mu_j$ is the utilization factor of the CPU core. The above expression relies on the approximation employed in [25] to estimate the mean response time of a $G/G/1$ queuing node when $c_{aj} \geq 1$.

Besides the CPU cores allocated to the VNFCs, let us assume that there are K additional resources allocated to the SNS. Each of these resources could also be modeled as a $G/G/1$ node, though this consideration does not affect the subsequent analysis and a completely different approach could be used instead. Under this consideration, the response time of the resource k, θ_k can be approximated as:

$$\theta_k = \frac{c_{sk}^2 + c_{ak}^2}{2} \cdot \frac{\rho_k}{\mu_k \cdot (1 - \rho_k)} + \frac{1}{\mu_k} \tag{4}$$

where parameters μ_k, c_{sk}^2, c_{ak}^2, and ρ_k are respectively the service rate, SCV of the service time, SCV of the inter-arrival times, and the utilization factor of the resource k.

The mean response time of the SNS $\overline{T}$ can be computed as:

$$\overline{T} = \sum_{j=1}^{J} V_j \cdot \overline{T}_j + \sum_{k=1}^{K} V_k \cdot \theta_k + T_{prop} \tag{5}$$

, where V_j and V_k respectively denote the visit ratio of the *VNFC* j and the resource k, and T_{prop} is a parameter to take into account the propagation delays. A visit ratio is defined as the average number of visits to a given node by an arbitrary job during its lifetime in the SNS.

For the sake of illustration, Fig. 2 shows an example of network service and a possible queuing theory model to capture its behavior. Specifically, for each VNF, the bottlenecks considered are the processor and the Network Interface Cards (NICs). Observe that we are assuming there is one NIC associated with each VNF exposed interface.

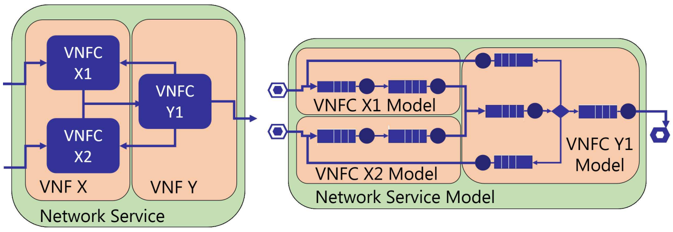

Fig. 2. Example of QT model for a given SNS.

V. Optimal Resource Dimensioning of SNSs

This section includes the closed-form expression for the VNFCs resources dimensioning, which was derived by using the method of Lagrange multipliers. The problem formulation and the SNS performance model considered are those respectively described in Sections III and IV.

Let $\mathscr{L}(m_i,\ldots,m_J,\gamma)$ denote the Lagrangian associated with the dimensioning problem formulated in Section III, which is given by $\mathscr{L}(m_i,\ldots,m_J,\gamma) = f(m_i,\ldots,m_J) + \gamma \cdot g(m_i,\ldots,m_J)$:

$$\mathscr{L}(m_j \forall j \in [1,J], \gamma) = \sum_{j=1}^{J} \alpha_j \cdot m_j + \gamma \cdot \left(\overline{T} - \overline{T}_{max}\right) \tag{6}$$

where γ is the Lagrange multiplier.

Corollary 1. *The resource dimensioning problem defined by* (1) *and* (2) *is convex considering the SNS performance model described in Section IV.*

Proof: The Hessian matrix of the Lagrangian $\nabla^2\mathscr{L}(m_i,\ldots,m_J,\gamma)$ is diagonal as $f(m_i,\ldots,m_J)$ and $g(m_i,\ldots,m_J)$ are given by the sum of J terms, each of which is a function of only one decision variable m_j of the problem. The jth element of the principal diagonal of $\nabla^2\mathscr{L}(m_i,\ldots,m_J,\gamma)$ is given by:

$$\nabla^2\mathscr{L}_{j,j} = \frac{\gamma \cdot V_j \cdot (c_{sj}^2 + c_{aj}^2) \cdot \rho_j}{\mu_j \cdot (m_j - \rho_j)^3} \tag{7}$$

The parameters c_{sj}^2, c_{aj}^2, ρ_j, and V_j are positive by definition[1] and γ is also positive as it can be observed in (12). Moreover, $m_j > \rho_j$, otherwise, the system would be unstable. Then, $\nabla^2\mathscr{L}$ is definite positive and the problem is convex. ■

Next, we can find the critical points m_j^* by solving $\nabla\mathscr{L}(m_j \forall j \in [1,J], \gamma) = 0$, where $\nabla\mathscr{L}$ denotes the gradient of the Lagrangian.

Theorem 1. *Considering the resource dimensioning problem defined by* (1) *and* (2) *and the SNS performance model presented in Section IV, the optimal number of CPU cores m_j^* to be allocated to each VNFC j of a given SNS so that its mean response time be lower than $\overline{T}_{max}$ is given by:*

$$m_j^* = \left\lceil \sqrt{\beta_j} \cdot \sum_{k=1}^{J} \alpha_k \cdot \sqrt{\beta_k} + \rho_j \right\rceil \tag{8}$$

where

$$\beta_j = \frac{V_j \cdot (c_{sj}^2 + c_{aj}^2) \cdot \rho_j}{2 \cdot \alpha_j \cdot \mu_j \cdot \left(\overline{T}_{max} - \theta - \sum_{k=1}^{J} \frac{V_k}{\mu_k}\right)} \tag{9}$$

[1] c_{sj}^2 and c_{aj}^2 can equal zero simultaneously in deterministic systems, but in such case the optimal solution of the problem can be computed easily

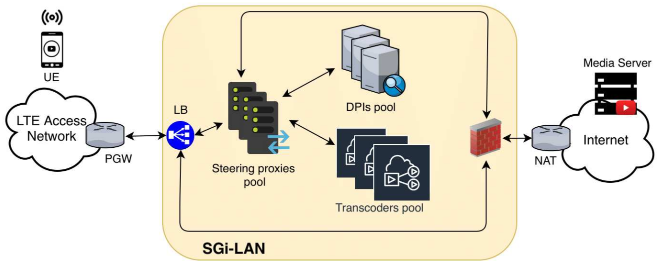

Fig. 3. The scenario considered in the experimental setup.

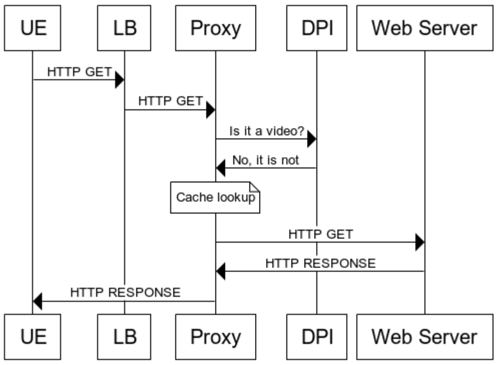

Fig. 4. Flow diagram for HTTP web traffic.

The above expressions allow us to determine the number CPU cores has to be allocated to each VNFC j of the SNS for minimizing the economic cost or the energy consumption, while the maximum response time of the SNS is guaranteed (i.e., $\overline{T} \leq \overline{T}_{max}$). Please refer to the appendix for the proof of Theorem 1.

VI. Results

A. Experimental Setup

We verified the correctness and usefulness of (8) for the scenario shown in Fig. 3. Specifically, we considered a typical Service Function Chain (SFC) deployed on the SGi-LAN of mobile networks for video optimization as described in [26]. The SFC consists of four essential service functions:

i) a Load Balancer (LB) that separates HTTP over TCP port 80 from the rest of traffic and distributes the HTTP load among a pool of proxies,

ii) a steering proxy that redirects HTTP traffic.

iii) a Deep Packet Inspector (DPI) which checks whether a given `HTTP GET` or `RESPONSE` is video content, and

iv) a Transcoder (XCDR) which converts videos to an appropriate format on the fly.

The flow diagram considered for requesting, downloading, and transcoding a given chunk of video, which is encapsulated in an `HTTP RESPONSE`, is depicted in [26, Fig. 8]. Figure 4 shows the flow diagram considered for processing non-video HTTP traffic.

We developed a simulator of the video optimization chain shown in Fig. 3 running in a virtualized infrastructure. The underlying physical infrastructure comprises 40 PMs or servers, each of which with 16 physical CPU cores, interconnected through a 10 Gbps Ethernet network with a tree topology and two layers of switches. The resources of the video optimization chain are adapted according to the demand by using the proposed solution.

TABLE I
Traffic model setup.

Web browsing	
Probability of a web session	0.99
HTTP GET Size	600 Bytes
HTTP RESPONSE Size	810 kBytes
Mean number of HTPP GETs per user session	230
Video streaming	
Probability of a video session	0.01
HTTP GET Size (Video)	800 Bytes
Mean HTTP RESPONSE Size before transcoding (Video)	131250 Bytes (1 second, 1280 × 720, 25 fps, H.264)
Mean HTTP RESPONSE Size after transcoding (Video)	38750 Bytes (1 second, 426 × 240, 25 fps, H.264)
Mean number of chunks per user session (Video)	231 (1 video per session, video duration distribution from [27])

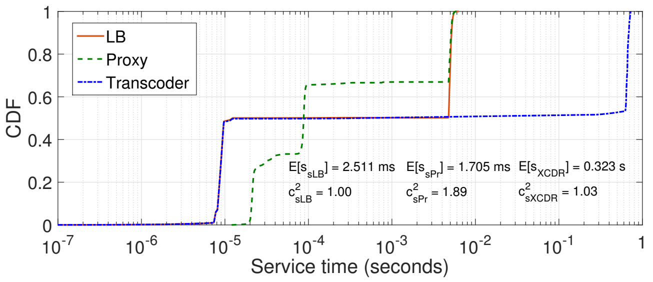

Fig. 5. Experimental service processes per CPU instance.

The workload was synthetically generated using the web browsing and HTTP video streaming traffic models, with some adaptations, included in [17]. Table I include a summary of the main characteristics of the traffic models considered.

Figure 5 depicts the Cumulative Distribution Functions (CDFs) of the LB, proxy, and XCDR service times. These curves were measured experimentally and represent the distribution of the service time required by a single processing instance to process a given HTTP message (`GET` or `RESPONSE`). Specifically, the processing instance considered in the setup was an Intel(R) Core(TM) i7-4600U CPU @ 2.10GHz. Figure 5 also includes the mean and SCV of the depicted CDFs.

Observe that the CDF of the LB, proxy, and XCDR service times present a ladder shape. This is because each service function type has to carry out different processing tasks depending on the kind of incoming HTTP message. For the sake of illustration, the XCDR acts as a proxy for the video `HTTP GET`s and performs transcoding for the video `HTTP RESPONSE`s. This fact explains that the XCDR service time distribution is a mixture of two different distributions, one associated with the lightweight processing of the `HTTP GET`s messages, and the other with the heavy processing of transcoding.

Regarding the DPI service process, it was considered deterministic ($c^2_{DPI} = 0$). The DPI service time per HTTP message was set to 3 ms considering that the DPI has to inspect the video `HTTP RESPONSE`s in the depth of 10 packets along with the measurements included in [28].

B. Resources Dimensioning Results

To validate the usefulness of (8), we carried out a set of simulations for eight different workloads. The workload is expressed as the incoming rate of HTTP GETs to the video optimization chain. The performance requirement considered was $\overline{T} \leq 30$ ms, which means that the video optimization chain has a budget of 30 ms on average to serve an HTTP message (e.g., `HTTP GET` or `HTTP RESPONSE`). The delay measurement of $4 \cdot 10^5$ HTTP messages was considered as stop condition for all the simulations. We observed that the system achieved convergence (steady-state) comfortably with this stop condition.

Table II includes the results of the above-mentioned set of simulations. The number of CPU cores allocated to each service function, which are included in the middle four columns of Table II, were computed using (8). To guarantee the correctness of the closed-form expression derivation, we verified that CVX, a package for specifying and solving convex programs [29], [30], achieved exactly the same results. The mean response time of the video optimization chain (see third column of Table II or Fig. 7) obtained for all the simulations is below $\overline{T}_{max} = 30$ ms, thus validating the usefulness of (8).

Fig. 7 depicts both the mean response time of the video optimization chain obtained by simulation (labeled as “Sim”) and predicted by the performance model described in Section IV (labeled as “Theo”). It also shows the relative error exhibited by the performance model (the same data are also included in Table II). The relative estimation error is below 18%, which is acceptable compared with the QT standard methodologies of analysis [31]. However, it should be noted that this error is not only due to the performance model itself, but also to the estimation error of the SCV of the aggregated arrival process to each service function (see columns eight to twelve in Table II and Fig. 8). Before launching a simulation, we knew neither the mean arrival rates or the SCV of the arrival processes. In a practical situation, there might be a workload predictor (see Fig. 1) which provides these parameters for a given time based on previous observations. To overcome this limitation of our setup, we estimated those parameters. The arrival rates are calculated easily and very accurately using the flow balance equations and the traffic model setup. For the estimation of the SCVs we used the methodology proposed in [25] considering that the number of CPU cores m_j allocated to each service function j equals $\lceil \lambda_j / \mu_j \rceil$ (stability condition).

Figure 6 shows the impact of the load and the value of the mean response time budget on the resources demand. As expected, the more stringent is the performance requirement, the more resources we need to allocate to the SNS to fulfill it. The most interesting result observed in Fig. 6 is that there is a point in the load where the demand of CPU cores shoots up. We observed that this is because of the rest of the resources consumed by the SNS start to exhibit a significant response time (congestion). Specifically, in our case, some of the links of the network that interconnects the PMs caused this behavior. This result highlights the importance of the integration and coordination of Software Defined Network (SDN) and NFV paradigms.

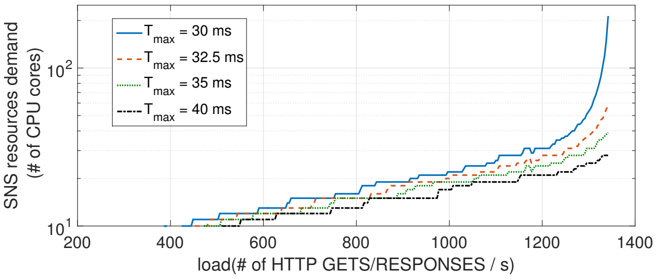

Fig. 6. The total number of CPU cores allocated to the video optimization chain versus the load for different values of T_{max}.

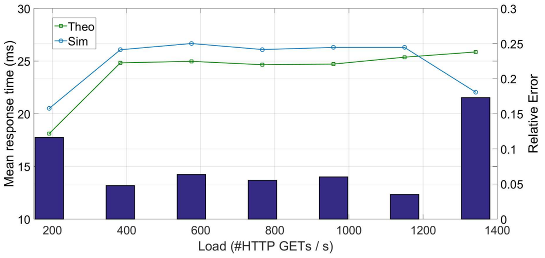

Fig. 7. SNS mean response time model validation.

VII. Conclusion

In this work, we have tackled the derivation of a closed-form expression for the optimal resources dimensioning of an SNS in terms of cost or energy efficiency. The performance requirement considered for the SNS is a limit on its mean response time. To estimate the performance of the SNS, we have considered a simple but practical and accurate queuing model. The usefulness of the derived closed-form expression has been successfully validated through simulation. The scenario considered for the validation is a video optimization chain located at the SGi-LAN of a mobile network.

Appendix

This Appendix includes the proof of the Theorem 1 (see Section V). The critical points m_j^* of the SNS resource dimensioning problem can be found by solving $\nabla \mathscr{L}(m_j \forall j \in [1,J], \gamma) = 0$, where $\nabla \mathscr{L}$ denotes the gradient of the Lagrangian. This yields the following set of nonlinear equations:

$$\begin{aligned} &\sum_{j=1}^{J} V_j \cdot \left(\frac{c_{sj}^2 + c_{aj}^2}{2} \cdot \frac{\rho_j}{\mu_j \cdot (m_j - \rho_j)} + \frac{1}{\mu_j} \right) - \overline{T}_{budget} \\ &\sum_{j=1}^{J} V_j \cdot \left(W_j + \frac{1}{\mu_j} \right) - \overline{T}_{budget} = 0 \end{aligned} \tag{10}$$

$$\begin{aligned} &\alpha_j - \gamma \cdot V_j \cdot \frac{c_{sj}^2 + c_{aj}^2}{2} \cdot \frac{\rho_j}{\mu_j \cdot (m_j - \rho_j)^2} \\ &\alpha_j - \gamma \cdot V_j \cdot \frac{W_j}{(m_j - \rho_j)} = 0 \quad \forall j \in [1,J] \end{aligned} \tag{11}$$

Where W_j denotes the mean waiting time at the CPU resource of the VNFC j. Then, solving (11) for W_j and substituting it in (10), and after solving (10) for γ, we get:

$$\gamma = \frac{\sum_{i=1}^{J} \alpha_i \cdot (m_i - \rho_i)}{\overline{T}_{budget} - \sum_{i=1}^{J} \frac{V_i}{\mu_i}} = \frac{\sum_{i=1}^{J} \alpha_i \cdot (m_i - \rho_i)}{\overline{T}_{max} - \theta - \sum_{i=1}^{J} \frac{V_i}{\mu_i}} \tag{12}$$

Then, by substitution of (12) in (11) and solving for $m_j \in \mathbb{N} \quad \forall j \in [1,J]$, we finally get (8) and (9). Last, under stability conditions and $\overline{T}_{max} \geq \theta - \sum_{i=1}^{J} V_i / \mu_i$, $\gamma > 0$ (see (12)). Then, considering Corollary 1, the Karush-Kuhn-Tucker (KKT) conditions are met and $m_j^* \forall j \in [1,J]$ is the global minimum of the problem, thus concluding the proof of Theorem 1.

TABLE II

VALIDATION RESULTS OF THE CLOSED-FORM EXPRESSION FOR THE RESOURCES DIMENSIONING OF A VIDEO OPTIMIZATION CHAIN.

Mean response time				Resources demand (# of CPU cores)				Estimation error SCV arrival processes			
Load (in HTTP GETs per second)	Theo.	Sim.	ε	LB	Proxy	DPI	XCDR	LB	Proxy	DPI	XCDR
192	18.1 ms	20.5 ms	11.6%	1	1	1	2	10.5%	4.1%	26.2%	8.9%
383	24.8 ms	26.1 ms	4.8%	3	2	1	3	88.1%	20.4%	56.7%	8.0%
575	25.0 ms	26.7 ms	6.4%	3	3	2	4	8.2%	14.9%	2.6%	22.81%
767	24.7 ms	26.1 ms	5.5%	4	5	2	6	16.7%	4.0%	35.3%	14.7%
958	24.7 ms	26.3 ms	6.0%	6	6	3	7	6.0%	3.8%	16.2%	9.6%
1150	25.4 ms	26.3 ms	3.5%	8	8	4	10	5.4%	10.1%	27.3%	21.5%
1342	25.9 ms	22.1 ms	17.3%	57	62	20	81	2.1%	2.8%	34.4%	35.6%

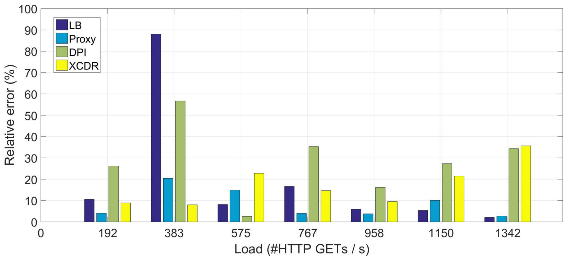

Fig. 8. Estimation error of the SCVs of the HTTP messages inter-arrival times at the different service functions.

ACKNOWLEDGMENT

This work is partially supported by the European Union's Horizon 2020 research and innovation programme under the MATILDA project and the Academy of Finland 6Genesis and CSN projects with grant agreement No. 761898, No. 318927, and No. 311654, respectively.

REFERENCES

[1] H. Hawilo, A. Shami, M. Mirahmadi, and R. Asal, "NFV: state of the art, challenges, and implementation in next generation mobile networks (vEPC)," *IEEE Network*, vol. 28, no. 6, pp. 18–26, Nov 2014.

[2] T. Taleb, "Toward Carrier Cloud: Potential, Challenges, and Solutions," *IEEE Wireless Communications*, vol. 21, no. 3, pp. 80–91, Jun. 2014.

[3] T. Taleb, M. Corici, C. Parada, A. Jamakovic, S. Ruffino, G. Karagiannis, and T. Magedanz, "EASE: EPC as a Service to Ease Mobile Core Network Deployment over Cloud," *IEEE Network*, vol. 29, no. 2, pp. 78–88, Mar. 2015.

[4] T. Taleb, A. Ksentini, and R. Jantti, "“Anything as a Service” for 5G Mobile Systems," *IEEE Network*, vol. 30, no. 6, pp. 84–91, Nov. 2016.

[5] T. Taleb, B. Mada, M. Corici, A. Nakao, and H. Flinck, "PERMIT: Network Slicing for Personalized 5G Mobile Telecommunications," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 88–93, May 2017.

[6] O. Adamuz-Hinojosa, J. Ordonez-Lucena, P. Ameigeiras, J. J. Ramos-Munoz, D. Lopez, and J. Folgueira, "Automated Network Service Scaling in NFV: Concepts, Mechanisms and Scaling Workflow," *IEEE Communications Magazine*, vol. 56, no. 7, pp. 162–169, Jul. 2018.

[7] B. Urgaonkar, P. Shenoy, A. Chandra, P. Goyal, and T. Wood, "Agile dynamic provisioning of multi-tier internet applications," *ACM Trans. Auton. Adapt. Syst.*, vol. 3, no. 1, pp. 1:1–1:39, Mar 2008.

[8] D. Huang, B. He, and C. Miao, "A Survey of Resource Management in Multi-Tier Web Applications," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1574–1590, Mar. 2014.

[9] R. Mijumbi, S. Hasija, S. Davy, A. Davy, B. Jennings, and R. Boutaba, "Topology-Aware Prediction of Virtual Network Function Resource Requirements," *IEEE Trans. Netw. Service Manag.*, vol. 14, no. 1, pp. 106–120, Mar. 2017.

[10] C. H. T. Arteaga, F. Rissoi, and O. M. C. Rendon, "An Adaptive Scaling Mechanism for Managing Performance Variations in Network Functions Virtualization: A Case Study in an NFV-based EPC," in *Proc. 2017 13th Int. Conf. on Netw. Service Manag. (CNSM)*, Tokyo, Japan, Nov. 2017.

[11] V. Eramo, A. Tosti, and E. Miucci, "Server Resource Dimensioning and Routing of Service Function Chain in NFV Network Architectures," *J. of Electrical and Computer Engineering*, vol. 2016, no. Article ID 7139852, pp. 1–12, 2016.

[12] L. Yala, P. A. Frangoudis, and A. Ksentini, "Qoe-aware computing resource allocation for cdn-as-a-service provision," in *2016 IEEE GLOBECOM*, Washington, DC USA, Dec 2016, pp. 1–6.

[13] M. S. Yoon and A. E. Kamal, "Nfv resource allocation using mixed queuing network model," in *2016 IEEE GLOBECOM*, Washington, DC USA, Dec 2016, pp. 1–6.

[14] V. Quintuna and F. Guillemin, "On dimensioning cloud-ran systems," in *Proc. 11th EAI Int. Conf. on Performance Evaluation Methodologies and Tools*, ser. VALUETOOLS 2017. New York, NY, USA: ACM, 2017, pp. 132–139.

[15] K. Tanabe, H. Nakayama, T. Hayashi, and K. Yamaoka, "An Optimal Resource Assignment for C/D-Plane Virtualized Mobile Core Networks," in *Proc. 2017 IEEE Int. Conf. on Commun. (ICC)*, Paris, France, May 2017.

[16] P. Andres-Maldonado, P. Ameigeiras, J. Prados-Garzon, J. J. Ramos-Munoz, and J. M. Lopez-Soler, "Virtualized mme design for iot support in 5g systems," *Sensors*, vol. 16, no. 8, 2016.

[17] J. Prados-Garzon, J. J. Ramos-Munoz, P. Ameigeiras, P. Andres-Maldonado, and J. M. Lopez-Soler, "Modeling and Dimensioning of a Virtualized MME for 5G Mobile Networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 5, pp. 4383–4395, May 2017.

[18] J. Prados-Garzon, A. Laghrissi, M. Bagaa, and T. Taleb, "A Queuing based Dynamic Auto Scaling Algorithm for the LTE EPC Control Plane," in *2018 IEEE GLOBECOM*, Abu Dhabi, Dec. 2018.

[19] L. Ruiz, R. J. Durn, I. De Miguel, P. S. Khodashenas, J.-J. Pedreo-Manresa, N. Merayo, J. C. Aguado, P. Pavn-Marino, S. Siddiqui, J. Mata, P. Fernndez, R. M. Lorenzo, and E. J. Abril, "A genetic algorithm for vnf provisioning in nfv-enabled cloud/mec ran architectures," *Applied Sciences*, vol. 8, no. 12, 2018.

[20] J. Prados-Garzon, A. Laghrissi, M. Bagaa, T. Taleb, and J. M. Lopez-Soler, "A Complete LTE Mathematical Framework for the Network Slice Planning of the EPC," *IEEE Trans. Mobile Comput.*, pp. 1–1, 2019.

[21] B. Ahmad, T. Taleb, A. Vajda, and M. Bagaa, "Dynamic Cloud Resource Scheduling in Virtualized 5G Mobile Systems," in *2016 IEEE Global Communications Conference (GLOBECOM)*, Washington, DC, USA, Dec. 2016.

[22] K. Tanabe, H. Nakayama, T. Hayashi, and K. Yamakoa, "vepc optimal resource assignment method for accommodating m2m communications," *IEICE Trans. on Commun.*, vol. advpub, 2017.

[23] L. Gu, D. Zeng, S. Tao, S. Guo, H. Jin, A. Y. Zomaya, and W. Zhuang, "Fairness-aware dynamic rate control and flow scheduling for network utility maximization in network service chain," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 5, pp. 1059–1071, May 2019.

[24] G. Tseliou, F. Adelantado, and C. Verikoukis, "Netslic: Base station agnostic framework for network slicing," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3820–3832, April 2019.

[25] W. Whitt, "The Queueing Network Analyzer," *The Bell System Technical Journal*, vol. 62, no. 9, pp. 2779–2815, Nov. 1983.

[26] W. Haeffner, J. Napper, M. Stiemerling, D. Lopez, and J. Uttaro, "Service function chaining use cases in mobile networks," Informational, IETF Secretariat, Internet-Draft draft-ietf-sfc-use-case-mobility-09.txt, Jan. 2019.

[27] J. J. Ramos-Muñoz, J. Prados-Garzon, P. Ameigeiras, J. Navarro-Ortiz, and J. M. López-Soler, "Characteristics of mobile youtube traffic," *IEEE Wireless Communications*, vol. 21, no. 1, pp. 18–25, 2014.

[28] M. . Liao, M. . Luo, C. . Yang, C. . Chen, P. . Wu, and Y. . Chen, "Design and evaluation of deep packet inspection system: A case study," *IET Networks*, vol. 1, no. 1, pp. 2–9, March 2012.

[29] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, Mar. 2014.

[30] ——, "Graph implementations for nonsmooth convex programs," in *Recent Advances in Learning and Control*, ser. Lecture Notes in Control and Information Sciences, V. Blondel, S. Boyd, and H. Kimura, Eds. Springer-Verlag Limited, 2008, pp. 95–110.

[31] J. Prados-Garzon, P. Ameigeiras, J. J. Ramos-Munoz, P. Andres-Maldonado, and J. M. Lopez-Soler, "Analytical modeling for Virtualized Network Functions," in *Proc. 2017 IEEE Int. Conf. on Commun. Workshops (ICC Workshops)*, Paris, France, May 2017, pp. 979–985.